Counting Arrays

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

VegiCat gives you two integers n and m. Consider integer arrays $a_1, a_2, \ldots, a_n$ such that

$$a_1 = 0, \qquad 0 \le a_{i+1} - a_i \le 1 \quad (1 \le i < n).$$

Let $s_i = a_1 + a_2 + \ldots + a_i$ be the prefix sums of a. VegiCat also tells you the prefix sums at m positions; that is, you are given m constraints of the form $s_{p_j} = v_j$ $(1 \le j \le m)$.

Count the number of arrays satisfying all constraints, modulo 998244353. If no array satisfies all the constraints, the answer is 0.

Input

The first line contains an integer T $(1 \le T \le 10^4)$, the number of test cases. The description of the test cases follows.

Each test case is given as follows:

The first line contains two integers n and m $(1 \le n \le 100,\ 0 \le m \le n)$, the length of the array and the number of prefix sum constraints, respectively.

The j-th of the next m lines contains two integers p_j and v_j $(1 \le p_j \le n,\ 0 \le v_j \le 10^9)$. This describes a constraint: the sum of the first p_j elements of the array must equal v_j.

The positions p_j are given in strictly increasing order.

Output

For each test case, output an integer — the number of valid arrays modulo 998244353.

Example

standard input	standard output
6 1 0 3 1 3 1 3 1 2 0 4 1 4 3 5 2 1 0 5 4 5 2 2 0 5 10	1 1 2 2 2 0

Note

Without any prefix constraints there are exactly 2^{n-1} such arrays, because each of the $n-1$ adjacent differences may independently be 0 or 1.

In the second test case, the only array with $s_3 = 1$ is $(0, 0, 1)$.

In the third test case, $s_2 = 0$ forces $a_2 = 0$, after which a_3 may be 0 or 1.

In the fourth test case, the arrays are $(0, 1, 1, 1)$ and $(0, 0, 1, 2)$.

In the fifth test case, the arrays with $s_5 = 4$ are $(0, 0, 1, 1, 2)$ and $(0, 1, 1, 1, 1)$.

In the sixth test case, $s_2 = 0$ forces $a_1 = a_2 = 0$, so $s_5 \leq 6$. The constraint $s_5 = 10$ is impossible.